A simple phase transition relaxation solver for liquid–vapor flows

Alexandre Chiapolino¹,², Pierre Boivin¹*,† and Richard Saurel¹,²,³

¹RS2N, Chemin de Gaumin, Saint-Zacharie 83640, France
²M2P2, Aix Marseille Université, CNRS, Centrale Marseille, M2P2 UMR 7340, Marseille 13451, France
³IUF, University Institute of France

SUMMARY

Determining liquid–vapor phase equilibrium is often required in multiphase flow computations. Existing equilibrium solvers are either accurate but computationally expensive or cheap but inaccurate. The present paper aims at building a fast and accurate specific phase equilibrium solver, specifically devoted to unsteady multiphase flow computations. Moreover, the solver is efficient at phase diagram bounds, where non-equilibrium pure liquid and pure gas are present. It is systematically validated against solutions based on an accurate (but expensive) solver. Its capability to deal with cavitating, evaporating, and condensing two-phase flows is highlighted on severe test problems both 1D and 2D. Copyright © 2016 John Wiley & Sons, Ltd.

Received 17 January 2016; Revised 4 May 2016; Accepted 18 July 2016

KEY WORDS: multiphase flows; interfaces; hyperbolic systems; relaxation; phase transition

1. INTRODUCTION

Most multiphase flow computations face phase transition modeling, and one of the difficulties is to adopt the correct mass transfer model, appropriate to a given situation. Some approaches deal with mixtures out of thermal and velocity equilibrium. When it is possible to determine the specific interfacial area separating the liquid and gas phases, determination of the mass transfer rate may be carried out on the basis of Nusselt and Sherwood correlations. Such a method was derived for spray evaporation by Ambramzon and Sirignano (1989) [1] and atmospheric flows by Jacobson (2005) [2]. Generalization to flashing and condensing sprays was carried out in Furfaro and Saurel (2016) [3].

However, determination of the specific interfacial area in two-phase mixtures is possible only for droplets and bubbly flows. When the topology is arbitrary, only limit case computations are possible, assuming the absence of mass transfer if the interfacial area is supposed to be very small, or assuming infinitely fast mass transfer (local thermodynamic equilibrium) if the interfacial area is supposed to be very large. When such an assumption is made, an appropriate equilibrium solver is needed.

The present paper deals with the building of such an equilibrium solver when non-equilibrium hyperbolic models such as Baer and Nunziato’s (1986) [4] are considered. However, the present method is not restricted to such a model but is also valid for its reduced versions such as the five-equation model of Kapila et al. (2001) [5] and its extension for cavitating flows, Saurel et al. (2008) [6], and Le Martelot et al. (2013) [7]. The homogeneous relaxation model (HRM) and homogeneous equilibrium model (Downar-Zapolski et al. (1996) [8], Barret et al. (2002) [9]) being also reduced versions of these models with respectively four and three equations, the present phase transition solver similarly applies to them.

*Correspondence to: Pierre Boivin, RS2N, Chemin de Gaumin, Saint-Zacharie 83640, France.
†E-mail: pierre.boivin@univ-amu.fr

Copyright © 2016 John Wiley & Sons, Ltd.
The theoretical link between these models was derived on the basis of asymptotic analysis in Saurel et al. (2008) [6] and more systematically by Lund (2012) [10]. Basically, the present phase transition solver may be used each time both liquid and gas compressibility are considered, as all formerly cited models consider this effect and this effect is responsible for their hyperbolic nature.

The five-equation, four-equation, and three-equation models are able to consider mixtures of fluids evolving respectively in mechanical, mechanical and thermal, and thermodynamic equilibrium. As they involve a single velocity (velocity disequilibrium is indeed absent), they are restricted mainly to specific applications such as follows:

- Cavitating flows, as it appears impossible in practice to address specific interfacial area determination and consequently model velocity slip. Computational examples of such flows are given for instance in Singhal et al. (2002) [11], Petitpas et al. (2009) [12], Le Martelot et al. (2013) [7], and Saurel et al. (2016) [13].
- Flashing and condensing flows, as they are high-speed flows and subject to stiff thermodynamic relaxation.
- Interfacial flows, as the same equations deal with the direct numerical simulation of boiling flows at sub-bubble scale (Le Martelot et al. (2014) [14], Saurel et al. (2016) [13]).

Therefore, the equilibrium solver addressed in the present work is a key point of the seven-equation, five-equation, four-equation, and three-equation hyperbolic two-phase flow models as it computes local thermodynamic equilibrium, this feature being important in many situations. The building of such an equilibrium solver has been addressed in Orbey et al. (1998) [15], Allaire et al. (2007) [16], Facchioni et al. (2012) [17], and Le Metayer et al. (2013) [18] on the basis of a highly nonlinear-algebraic model made from the saturation conditions, mixture mass, and mixture energy definitions. This system may cause difficulties as a result of nonlinearities and single-phase bounds of pure liquid and pure vapor, where it becomes ill posed.

In the present paper, a novel approach is promoted where the solution relaxes weakly (smoothly during time evolution) to the correct solution, on the basis of some estimates. After providing the background and context of the model in Sections 2 and 3, the two main ideas constituting the present model are successively detailed in Sections 4 and 5:

- Limitation of the relaxation term, following a Minmod-type procedure, reminiscent of slope limiters in high-order hyperbolic solvers (Van Leer et al. (1979) [19]).
- Approximate analytical estimation of the pressure at thermodynamic equilibrium.

These two treatments lead to a faster procedure than the usual iterative process: Reported computation times can be halved with the new algorithm. In addition, the algorithm described hereafter presents a very simple implementation, which is also a significant improvement over iterative procedures. Indeed, the method properly deals with limit cases, when only liquid or vapor is present, free to evolve out of the saturation conditions. The last sections (Sections 6, 7, and 8) are dedicated to illustrating the capabilities of the method, with several computational 1D and 2D examples of flows with evaporation, condensation, cavitation in the bulk fluid, and at liquid–vapor interfaces.

2. FLOW MODEL

The phase transition relaxation solver may be used with models mentioned previously (with seven, five, four, and three partial differential equations), but its presentation is simplified in the context of the four-equation model (often called HRM) as the solver directly connects the four-equation and three-equation models. When dealing with more sophisticated formulations, for instance, the seven-equation model, extra ingredients have to be presented, such as velocity and pressure relaxation solvers (for example, [20]). Here, there is a single step that makes the connection from the four-equation model (modeling mixtures out of thermodynamic equilibrium) and the three-equation model (mixtures in full equilibrium). Therefore, for the sake of simplicity, the four-equation model (HRM) is considered in the present paper as starting point. The corresponding flow model reads
on the basis of the stiffened gas EOS, an improved formulation the phase diagram. The building of such EOS has been addressed in Le Métayer representsthemixturecenterofmassvelocity; \( \mathbf{u} \) density of the liquid (\( l \) subscript) and gas (\( g \) subscript) phases. \( \rho \) represents the mixture density, and \( \mathbf{u} \) represents the mixture center of mass velocity; \( p \) denotes the mixture pressure, and \( E \) denotes the mixture total energy (\( E = e + u^2/2 \)). The mixture internal energy is defined as \( e = Y_l e_l + Y_g e_g \). Mass transfer has been omitted in System (1) as it is addressed later.

System (1) is clearly reminiscent of the reactive (or multicomponent) Euler equations widely used in chemically reacting flows. However, the thermodynamic closure differs significantly from the one used in gas mixtures because each phase is assumed to occupy its own volume. Indeed the mixture equation of state (EOS) is a consequence of the following algebraic system:

\[
\begin{align*}
T_l &= T_g = T, \\
e &= Y_l e_l(T, p) + Y_g e_g(T, p), \\
p_l &= p_g = p, \\
v &= Y_l v_l(T, p) + Y_g v_g(T, p),
\end{align*}
\]

where \( v_l, v_g, \) and \( v \) are respectively the specific volumes of the liquid, gas, and mixture.

In this frame, both liquid and gas require their own EOS, with parameters carefully chosen to fit the phase diagram. The building of such EOS has been addressed in Le Métayer et al. (2004) [21], on the basis of the stiffened gas EOS, an improved formulation NASG) being available as well [22].

The main formulas for the stiffened gas EOS read for a given phase \( k = l, g \),

\[
\begin{align*}
T_k(p_k, \rho_k) &= T_k(p_k, \rho_k), \\
p_k(p_k, \rho_k) &= p_k + p_{\infty,k}, \\
g_k(p_k, T_k) &= \left( \frac{\gamma_k}{1} \right) - \gamma_k p_{\infty,k} T_k, \\
e_k(p_k, T_k) &= \left( \frac{\gamma_k}{1} \right) - \gamma_k p_{\infty,k} T_k \ln \left( \frac{T_k^{\gamma_k}}{p_k + p_{\infty,k}} \right) + q_k,
\end{align*}
\]

where the following parameters are needed for each phase: \( \gamma_k, p_{\infty,k}, C_{v,k}, q_k, \) and \( q'_{k} \). From Eqs. (3), two other equations are found,

\[
\begin{align*}
v_k(p_k, T_k) &= \frac{\gamma_k C_{v,k} T_k}{p_k + p_{\infty,k}}, \\
e_k(p_k, T_k) &= \frac{\gamma_k}{p_k + p_{\infty,k}} + q_k.
\end{align*}
\]

As shown in [21], there is no difficulty to obtain these parameters once the saturation curves \( (p_{sat}(T), v_{g,sat}(T), v_{l,sat}(T), h_{g,sat}(T), h_{l,sat}(T)) \) are known. Saturation pressure and temperature obeying formulation (3) are linked through the saturation curve,
asthepreviousrelations(7)arefullyexplicitforthemixture[13,14].Themixturetemperature,energy,andpressure,thatcorrespondtothemixtureEOS:

\begin{equation}
\ln(p_{sat} + p_{\infty,g}) = A + \frac{B}{T_{sat}} + C \ln(T_{sat}) + D \ln(p_{sat} + p_{\infty,l}),
\end{equation}

with,

\begin{equation}
A = \frac{C_{p,l} - C_{p,g} + q_l' - q_l}{C_{p,g} - C_{v,g}}, \quad B = \frac{q_l - q_g - q_l}{C_{p,g} - C_{v,g}}, \quad C = \frac{C_{p,l} - C_{p,g}}{C_{p,g} - C_{\infty,g}}, \quad D = \frac{C_{p,l} - C_{v,l}}{C_{p,g} - C_{v,g}},
\end{equation}

which is equivalent to equaling the Gibbs free energies \(g_l\) and \(g_g\) from the stiffened gas EOS (3) (see [21] for details).

For liquid water and steam, the fluid parameters, optimized in the [298–473 °C] temperature range are given in Table I.

These parameters are used in the computational examples of the present paper. With the mixture thermodynamic closure (2), it is straightforward to derive the following analytical relations for the mixture temperature, energy, and pressure, that correspond to the mixture EOS:

\begin{equation}
\begin{cases}
T = T(v, p, Y_l), \\
e = e(T, p, Y_l), \\
p = p(v, e, Y_l),
\end{cases}
\end{equation}

The thermodynamic closure presented in [21] or its improved formulation [22] is very convenient, as the previous relations (7) are fully explicit for the mixture [13, 14]. The mixture temperature reads

\begin{equation}
T(v, p, Y_l) = v \left( \frac{(Y_l - 1)Y_l C_{v,l}}{p + p_{\infty,l}} + \frac{(Y_g - 1)Y_g C_{v,g}}{p + p_{\infty,g}} \right)^{-1};
\end{equation}

the mixture internal energy reads

\begin{equation}
e(T, p, Y_l) = Y_l \left( C_{v,l} T \frac{p + Y_l p_{\infty,l}}{p + p_{\infty,l}} + q_l \right) + Y_g \left( C_{v,g} T \frac{p + Y_g p_{\infty,g}}{p + p_{\infty,g}} + q_g \right),
\end{equation}

and the pressure reads

\begin{equation}
p(v, e, Y_l) = \frac{1}{2} (A_l + A_g - (p_{\infty,l} + p_{\infty,g})) + \sqrt{\frac{1}{4} (A_g - A_l - (p_{\infty,g} - p_{\infty,l}))^2 + A_l A_g},
\end{equation}

with

\begin{equation}
A_k = \frac{Y_k (Y_k - 1) C_{v,k}}{(Y_l C_{v,l} + Y_g C_{v,g})} \left( \frac{e - (Y_l q_l + Y_g q_g)}{v} - p_{\infty,k} \right),
\end{equation}

where the subscript \(k\) denotes liquid (l) and gas (g) phases (see [14] for details).

Albeit the apparent simplicity of the thermodynamic closure chosen, the phase transition model presented here may be extended to other thermodynamic closures given that each phase EOS is convex. Such extension is immediate with NASG [22].
Figure 1. Schematic representation in a (x,t) diagram of the three waves present in the flow model (1).

The reactive Euler equations govern the propagation of three waves throughout space (Figure 1). The middle wave (travelling along $u$) is a contact discontinuity, while the left and right waves (travelling along $u \pm c$, $c$ being the speed of sound) are nonlinear acoustic waves and can be either shocks or rarefactions.

With the thermodynamic closure (2), System (1) is hyperbolic with wave speeds $u$, $u + c$, and $u - c$. The sound speed for this system is given in Le Martelot et al. (2014) [14],

$$c^2 = \frac{1}{2} \left[ e - (Y_l q_l + Y_g q_g) \right] \left( a_1 + a_2 \right) + \frac{1}{2} \left( \frac{\partial R_l}{\partial \rho} \right) e R_1 + \frac{1}{2} \left( \frac{\partial R_2}{\partial e} \right) e \sqrt{\frac{1}{4} R_1 R_1 + R_2}$$

(12)

This sound speed can then be compared with a simpler approximation of the sound speed given by Wood [23]:

$$\frac{1}{\rho c^2} = \frac{\alpha_l}{\rho_l c_l^2} + \frac{\alpha_g}{\rho_g c_g^2}$$

(14)

As shown in Figure 2, Wood’s expression for the sound speed (Eq. (14)) is always slightly greater than the sound speed given by Eq. (12). It is thus more convenient (and simpler) for computational purposes related to the hyperbolic solver.

3. PHASE TRANSITION MODEL

When phase transition is addressed in System (1), the equations for the mixture mass, momentum and energy are unaffected, and only the mass fraction equation is modified through Gibbs free energy relaxation terms,
Figure 2. The speed of sound of System (1) given by Eq. (12) is compared with Wood’s sound speed (Eq. (14)). Full view and close up. The thick lines represent Wood’s sound speed. The dashed lines represent the augmented Euler’s sound speed. The two-phase mixture is made of liquid water and air at atmospheric conditions.

Figure 3. Representation of a control volume in the flow model during the phase transition step.

\[
\frac{\partial \rho Y_l}{\partial t} + \rho \frac{\partial u Y_l}{\partial x} = \rho v (g_g - g_l) \tag{15}
\]

where \( g_k \) denotes the phase k Gibbs free energy. \( v(A_f, T, p) \) represents a relaxation parameter that controls the rate at which thermodynamic equilibrium is reached. It is a function of the interfacial area \( A_f \), temperature, and pressure. Its determination is possible only when the interfacial area \( A_f \) is available, as with droplets and bubbly flows. In the present paper work, \( v \) is considered very big, so that relaxation to thermodynamic equilibrium is immediate.

Using a fractional step method, phase transition is decoupled of transport and wave propagation. At each time-step of the flow solver, the following equation has to be resolved for the mass fraction \( Y_l \),

\[
\frac{\partial (\rho Y_l)}{\partial t} = \rho \frac{(Y_l^* - Y_l)}{\tau} \tag{16}
\]

where \( Y_l^* \) is the liquid mass fraction at thermodynamic equilibrium. This equation trivially solves to a solution exponentially tending to \( Y_l^* \), with a characteristic time \( \tau \). In this work, \( \tau \) is assumed to be smaller than the other characteristic times of the flow model: Stiff relaxation is considered. As a consequence, solving Eq. (15) at every time-step reduces to setting \( Y_l = Y_l^* \) after each hyperbolic step.

Although the specific volume \( v = 1/\rho \) and energy \( e \) do not vary in the mixture during thermodynamic relaxation, the pressure and temperature do, also reaching their equilibrium values \( (p^*, T^*) \) on the saturation curve, because \( g_l = g_g^* \) is equivalent to the saturation condition: Eq. (5). The phase transition model thus reduces to computing the equilibrium state \( (p^*, T^*, v, e, Y_l^*) \), at every time step, from the state described by \( (p, T, v, e, Y_l) \), as represented schematically in Figure 3.
With the thermodynamic closure (2) presented earlier, the equilibrium state satisfies

\[
\begin{align*}
  p^*(v, e, Y_l^*) &= p_{\text{sat}} \left( T^*(v, e, Y_l^*) \right), \\
  v &= Y_l^* v_l(T^*, p^*) + (1 - Y_l^*) v_g(T^*, p^*), \\
  e &= Y_l^* e_l(T^*, p^*) + (1 - Y_l^*) e_g(T^*, p^*),
\end{align*}
\]

(17)

unless there is a solution in which the mixture is a pure phase (resp. \( Y_l^* = 0 \) or \( Y_l^* = 1 \)), with a temperature respectively above or below the saturation temperature. The previous nonlinear system can be solved following an iterative algorithm such as of Le Metayer et al. (2013) [18] summarized in Appendix A, but the aim of this paper is to offer a simpler and faster alternative.

4. THERMOCHEMICAL RELAXATION ALGORITHM

Let us remind that with an iterative approach, the goal of the thermochemical relaxation is to compute accurately \( Y_l^* \), the liquid mass fraction at equilibrium (or alternatively \( Y_g^* \)), while with the present method, the aim is to reach the same solution but gradually (typically two or three time steps).

The first step is to check with pure fluids existence. For numerical reasons, pure fluid conditions are considered via \( \epsilon \to 0 \), typically on the order of \( 10^{-8} \). To do this, \( Y_l^* = \epsilon \) and \( Y_l^* = 1 - \epsilon \) are successively assumed. Under these assumptions, the pressures are computed by use of the equation of state for the mixture (7),

\[ p = p(v, e, Y_l) \]

because \( v \) and \( e \) are invariant through phase transition. The associated temperatures are then computed through Eq. (8),

\[ T = T(v, p, Y_l) \]

The corresponding temperatures are compared with the saturation one at the current pressure (\( T_{\text{sat}}(p) \)),

\[
\begin{align*}
  \text{if } (Y_l = \epsilon \text{ and } T > T_{\text{sat}}) \text{ then } Y_l^* &= \epsilon \text{ (overheated vapor)}, \\
  \text{if } (Y_l = 1 - \epsilon \text{ and } T < T_{\text{sat}}) \text{ then } Y_l^* &= 1 - \epsilon \text{ (subcooled liquid)}. 
\end{align*}
\]

(18)

If one of the two inequalities is fulfilled the equilibrium liquid mass fraction, \( Y_l^* \) is fully determined, and no further computation is required. If none of the previous statements is true, then necessarily,

\[ \epsilon < Y_l^* < 1 - \epsilon, \quad p^* \text{ is unknown}, \]

(19)

and System (17) has to be resolved. The difficulty resides in the non-trivial relationship between the saturation pressure and saturation temperature arising from Eq. (5):

\[ T_{\text{sat}}(p) = -\frac{B}{C \cdot W \left( -\frac{Be^{A/C} p^{-1/C} (p + p_{\infty,C})^{D/C}}{C} \right)} \]

(20)

where \( W \) is the Lambert W function\(^4\), which cannot be expressed analytically, calling for an iterative method such as Newton’s.

The idea of the algorithm is to start from a rough estimate of the equilibrium pressure \( p^* = p \), and the associated equilibrium temperature \( T^* = T_{\text{sat}}(p) \) and reach gradually the solution. Because the pressure and temperature are related at saturation, the liquid internal energy \( e_l(p, T) \) and specific...

\(^4\)The Lambert W function is defined as \( z = W(ze^z) \).
volume \( v_l(p, T) \) become two functions depending of \( p \) only, and two values for \( Y_l^* \) as functions of the initial pressure \( p \) are obtained from System (17), by either using the mixture mass definition,

\[
Y_l^m(p) = \frac{v - v_g(p)}{v_l(p) - v_g(p)},
\]

or the mixture internal energy definition,

\[
Y_l^e(p) = \frac{e - e_g(p)}{e_l(p) - e_g(p)}.
\]

Indeed, these two formulas are equal only if \( p \) is exactly the equilibrium pressure \( p^* \), which is not the case \textit{a priori} because the process is not isobaric.

Based on these two guesses, a strategy inspired by flux limiters used in high-order schemes [19] is adopted. Let us introduce the ratio of the liquid mass fractions variation induced by the mass \((Y^m - Y)\) and internal energy \((Y^e - Y)\) guesses for the equilibrium mass fraction,

\[
r = \frac{Y_l^m(p) - Y_l}{Y_l^e(p) - Y_l}.
\]

An estimate of the equilibrium mass fraction is then obtained as

\[
\begin{cases}
    \text{if } r < 0, & Y_l^* = Y_l^m, \\
    \text{if } 0 < r < 1, & Y_l^* = Y_l^\text{mass}, \\
    \text{if } r > 1, & Y_l^* = Y_l^\text{energy}.
\end{cases}
\]

Alternatively, it also expresses as

\[
r = \left(\frac{Y_l^m(p) - Y_l}{Y_l^e(p) - Y_l}\right),
\]

\[
\begin{cases}
    \text{if } r < 0, & Y_l^* = Y_l \\
    \text{otherwise} & Y_l^* = Y_l + \text{sgn} \left[ Y_l^m(p) - Y_l \right] \times \text{Min} \left[ |Y_l^m(p) - Y_l|, |Y_l^e(p) - Y_l| \right].
\end{cases}
\]

In the first case, the evolutions indicated by the two equilibrium guesses are discordant: One tends to evaporate, whereas the other tends to condensate. The four quantities \((Y_l, Y_l^*, Y_l^m, Y_l^e)\) are then likely to be very close from one another, and no mass transfer is to be considered. In the other case, among \((Y_l^m, Y_l^e)\), the closest to the initial value \(Y_l\) is to be chosen. In that sense, the algorithm ensures equality of Eqs. (21) and (22) in the weak sense, rather than in the strong sense. Negative mass fractions are impossible to obtain with this algorithm. This can be seen from Figures 4, 5, and 7: Slopes of \(Y_l^e\) and \(Y_l^m\) are of different sign. Because the crossing of the two lines occurs at a positive mass fraction (which is the exact solution), only one of \(Y_l^e\) and \(Y_l^m\) can be negative at a time. Following the algorithm, there are then two possibilities: if the initial \(Y_g\) (from the hyperbolic step) is between \(Y_l^e\) and \(Y_l^m\), then nothing happens thanks to the Minmod-like limitation. If \(Y_g\) is not between \(Y_l^e\) and \(Y_l^m\), then it is necessarily above the maximum of the two (because \(Y_g > 0\)), and the algorithm will automatically pick the closest estimation (which is then positive).

The reason for this algorithm efficiency is illustrated in Figure 4, which presents the evolution of \(Y_l^m\) and \(Y_l^e\) as functions of the initial guess for the pressure, for a mixture initially away from thermodynamic equilibrium. It is seen that following the previous algorithm, which in the depicted case returns \(Y_l^* = Y_l^e\), gives a result within half a percent of the exact value, even though the initial state is quite far from equilibrium: 30 K below the saturation temperature at the initial pressure of 1 atm.
Figure 4. Evolution of $Y_{ml}$ and $Y_{e1}$ with $p$ (thick dashed and dashed-dotted lines resp.) corresponding to Eqs. (21) and (22). The exact values of $Y_{ml}$ and $p^*$ are found at the crossing of the two lines (•). Initial state (•): $p = 1\text{ atm}, Y_l = 0.2, T = T_{sat} = 30\text{ K} = 343\text{ K}$. The triangles represent the value of $Y_{ml}$ (▲) and $Y_{e1}$ (▼) evaluated by the algorithm. Relative error between the exact solution $Y_{ml}$ and that obtained with the algorithm is indicated.

Figure 5. Evolution of $Y_{mg}$ and $Y_{eg}$ from Eqs. (21) and (22) near the pure liquid bound with $p$ (thick dashed and dashed-dotted lines resp). The exact values of $Y_{mg}$ and $p^*$ are found at the crossing of the two lines (•). Initial state (•): $p = 1\text{ atm}, Y_g = 0.001, T = T_{sat} = 30\text{ K} = 403\text{ K}$. The triangles represent the value of $Y_{mg}$ (▲) and $Y_{eg}$ (▼) evaluated by the algorithm. a) $p$ is directly used for the initial guess of $p^*$. b) Based on the improved guess for $p^*$. Relative error between the exact solution $Y_{mg}$ and that obtained with the algorithm is indicated.

5. FURTHER IMPROVEMENT: ESTIMATE OF THE EQUILIBRIUM PRESSURE $p^*$

When the initial conditions are such that $p$ is far from the equilibrium pressure $p^*$ or, equivalently, that a small change in $Y_l$ creates an important pressure variation, the previous algorithm may fail. Because the liquid has a low compressibility, this will be typically the case when $Y_l$ is close to 1 (or, equivalently $Y_g \ll 1$). A small variation of the liquid mass fraction $Y_l$ when close to unity involves large variations of pressure. Figure 5a shows such a behavior: Starting from a small gas mass fraction $Y_g = 1 - Y_l = 10^{-3}$ and far from equilibrium ($T = T_{sat} + 30\text{ K}$), the algorithm previously described would compute $Y_m^* = 1 - Y_l^*$ as the equilibrium mass fraction. Although the mass transfer obtained has the proper sign, which is essential to the method convergence, it is seen in Figure 5a that $Y_g^*$ is underestimated by 77%. This section aims at improving this estimate, while keeping a low computational cost.
As it is usually the case when looking for an approximate solution, it is convenient to identify the bounds for the equilibrium pressure \( p^* \). Let us study the variations of the equilibrium pressure \( p^* \) as a function of an initial set of conditions \( (p, T, v, e, Y_g) \), where \( T \leq T_{sat}(p) \). The variation of the equilibrium pressure with the initial gas mass fraction \( Y_g \) is shown in Figure 6, for an initial pressure of 1 atm, and a temperature 30 K over the corresponding saturation temperature \( T = 403 \) K. As expected, the equilibrium pressure variation is monotonic. The extrema of \( p^* \) correspond necessarily to its value in the limits \( Y_g \rightarrow 0 \) and \( Y_g \rightarrow 1 \).

In the limit \( Y_l \rightarrow 1 \), the mixture is mostly liquid. Mass transfer between the two phases does not modify the temperature significantly, as the energy is mainly the one of the liquid that has the highest internal heat capacity. However, a small change in \( Y_g \) will lead to a significant pressure change. Saturation conditions lead to \( T = T_{sat}(p) \) and \( p = p_{sat}(T) \): (25)

\[
2 \leq \frac{p\cdot T_{sat}(p)}{T}.
\]

(26)

The following step is to estimate \( Y_l^m(p) \) and \( Y_l^e(p) \) at the two equilibrium pressure limits \( (p_1^*, p_2^*) \) defined by Eqs. (25) and (26), according to Eqs. (21) and (22). The corresponding points are shown in the Figure 7.

Assuming \( Y_l^m(p) \) and \( Y_l^e(p) \) to be linear functions of \( p \) between \( p_1^*, p_2^* \), it is straightforward to obtain an approximate expression for the equilibrium pressure found at the intersection of \( Y_l^m(p) \) and \( Y_l^e(p) \):

\[
p^* = (1 - \theta) p_1^* + \theta p_2^*.
\]

(27)

where,

\[
\theta = \frac{Y_l^m(p_1^*) - Y_l^e(p_2^*)}{(Y_l^m(p_1^*) - Y_l^e(p_2^*)) - (Y_l^m(p_1^*) - Y_l^e(p_2^*)).}
\]

(28)

Figure 6 compares the approximate equilibrium pressure with its exact value, showing that the maximum error is of the order of 5%. Note that these results are obtained for a mixture far from
Figure 7. Evolution of $Y_m^g$ and $Y_e^g$ from Eqs. (21) and (22) with $p$ (thick dashed and dashed-dotted lines resp.). The exact values of $Y_m^g$ and $p^*$ are found at the crossing of the two lines (●). The approximated value of $p^*$ is found at the crossing of the two straight lines. Initial state (●): $p = 1$ atm, $Y_g^e = 0.05$, $T = T_{sat} + 30$ K = 403 K. The triangles represent the value of $Y_m^m$ (▲) and $Y_e^e$ (▼) evaluated by the algorithm.

equilibrium initially: $T = T_{sat}(p) + 30$ K. In practice, the algorithm does not have to deal with mixture that far from equilibrium, as it is used at each time-step of the computation.

Following the algorithm presented in the previous section, but based on the initial guess for the pressure given by Eq. (27), the estimated value for the equilibrium mass fraction $Y_g^e$ is seen in Figure 5b to be considerably improved, down to an error of about 0.5%. The resulting final algorithm is summarized hereafter.

Summary of the procedure:

Input : $p$, $\rho$, $T$, $Y_l$ or $Y_g$ from the hyperbolic step.
Output : $Y_l^e$ or $Y_g^e$ the equilibrium mass fractions.

1) Search of pure fluid thermodynamic conditions

- Limit cases $Y_l^e \rightarrow \epsilon$ and $Y_g^e \rightarrow \epsilon$ are considered with computation of the pressure and temperature according to the mixture equation of state (8) and (10).
- If either subcooled liquid or superheated thermodynamic conditions are found according to Eq. (18), then $Y_l^e$ or alternately $Y_g^e$ is either $\epsilon$ or $1 - \epsilon$.

2) Otherwise, the thermodynamic equilibrium must be found

a) Estimation of the equilibrium pressure

- Calculation of $p_l^*$ according to Eq. (25) using the temperature from the hyperbolic step.
- Calculation of $p_g^*$ according to Eq. (26) using the temperature and pressure from the hyperbolic step.
- Prediction of the equilibrium pressure $p^*$ according to Eq. (27) using Eqs. (4), (21), (22) and (28). The mixture internal energy and specific volume come from the hyperbolic step.

b) Equilibrium computation

- Computation of the liquid mass fraction at equilibrium $Y_l^e$ according to the Minmod-like procedure (Eqs. (23) and (24)) presented in Sec. 3 using Eqs. (4), (21) and (22) estimated at $p^*$ and $T^* = T_{sat}(p^*)$. 

6. RESULTS

In order to illustrate the robustness and the accuracy of the relaxation algorithm, a shock tube containing liquid water and its own vapor is considered, with variable initial conditions. Shock tube tests appear as excellent benchmarks as the flow contains shock waves, contact discontinuities, and rarefaction fans that create some arduous conditions. In this section, the tube is 1 m long, and the initial discontinuity is located at 0.5 m. Boundary conditions are considered as non-reflecting. The computations that follow are addressed with the first-order Godunov method and the HLLC Riemann solver (see Toro (1997) [24], Saurel et al. (2016) [13], for details). Obviously, higher-order extensions can be considered, but this is not the scope of the present work.

The initial conditions correspond to both saturated liquid and vapor each time a mixture is present initially. Given initial pressures and mass fractions, the initial temperatures are computed with Eq. (5), and initial mixture energy and specific volumes are deduced from the definitions given by Eq. (2).

This section presents the results obtained with the present relaxation solver and a classic root-finding procedure method such as Newton’s.

6.1. Shock tube test with a mixture containing mainly water vapor

A two-phase mixture with an initial vapor mass fraction of 0.8 is considered throughout the entire tube with an initial pressure ratio of 2, resulting in the presence of initial density and temperature discontinuities. The results are shown at time $t \approx 0.8 \text{ ms}$ in Figure 8.

![Graphs showing the results of shock tube test](image)

Figure 8. Comparison of the present relaxation algorithm (thick lines) versus the nonlinear Newton’s method (symbols). Shock tube test with a two-phase mixture mainly made of water vapor. The dotted lines represent the solutions without phase transition. The dashed lines represent the initial conditions: $p_l = 2 \cdot 10^5 \text{ Pa}$, $p_r = 10^5 \text{ Pa}$, $u_l = u_r = 0 \text{ m s}^{-1}$, $Y_l^{liq} = Y_r^{liq} = 0.2$. Final time: $t \approx 0.8 \text{ ms}$. Mesh: 100 cells. For the sake of visualization, only 50 symbols out of 100 are plotted for the nonlinear Newton’s method.
A SIMPLE PHASE TRANSITION RELAXATION SOLVER FOR LIQUID–VAPOR FLOWS

Figure 9. Comparison of the present relaxation algorithm (thick lines) versus the nonlinear Newton’s method (symbols). Shock tube test with a two-phase mixture involving a vanishing liquid phase. The dotted lines represent the solutions without phase transition. The dashed lines represent the initial conditions: $p_l = 2 \cdot 10^5$ Pa, $p_r = 10^5$ Pa, $u_l = u_r = 0$ m s$^{-1}$, $Y^{liq}_l = Y^{liq}_r = 0.01$. Final time: $t = 0.5$ ms. Mesh: 100 cells. For the sake of visualization, only 50 symbols out of 100 are plotted for the nonlinear Newton’s method.

Perfect agreement is obtained between the present relaxation solver and the nonlinear iterative method recalled in Appendix A. Clearly the shock compression yields evaporation, and the rarefaction results into condensation. Note that the initial left and right sides are now connected with the same temperature in the central area of the tube. This temperature being the saturation temperature at the current pressure denotes as $T^* = T_{sat}(p^*)$. Obviously, without phase transition, two different temperatures evolve at the central region, separated by a contact wave.

6.2. Shock tube test with a vanishing liquid phase

We now address a similar test as the previous one but with a vanishing phase. The initial liquid mass fraction is 0.01 throughout the entire tube. The results are shown at time $t = 0.5$ ms in Figure 9.

Perfect agreement between the two solvers is again obtained. Here, the shock compression of the liquid–gas mixture yields total evaporation. Oppositely, the expansion wave yields condensation. Appearance of pure gas is computed without oscillations by the two methods.

6.3. Shock tube test with a mixture containing mainly liquid water

A two-phase flow with an initial liquid mass fraction of 0.8 throughout the entire tube is now considered with the same pressure ratio as previously mentioned. The results are shown at time $t = 1.5$ ms in Figure 10.
A condensation process is now observed at the shock while evaporation (cavitation) appears through the rarefaction fan. The agreement between the two methods is again excellent.

6.4. Shock tube test with a vanishing gas phase

The behavior of the same methods is now examined with a vanishing vapor phase. The same two-phase flow shock tube is addressed with an initial liquid mass fraction of 0.999 throughout the entire tube. The results are shown at time $t \approx 8$ ms in Figure 11.

Condensation again appears towards the shock and evaporation through expansion waves. Perfect agreement between the two methods is again observed. Pure liquid appearance is obtained without oscillations.

6.5. Superheated vapor

Considering pure fluids at initial states, it is interesting to set initial conditions away from the thermodynamic equilibrium. Superheated pure vapor is considered throughout the tube. The temperature is therefore above the saturation temperature and is set at 450 K. The pressure is set at $10^5$ Pa. The initial velocity is set at $-200$ m s$^{-1}$ at left and $+200$ m s$^{-1}$ at right. The results are
A SIMPLE PHASE TRANSITION RELAXATION SOLVER FOR LIQUID–VAPOR FLOWS

Figure 11. Comparison of the present relaxation algorithm (thick lines) versus the nonlinear Newton’s method (symbols). Shock tube test with a two-phase mixture involving a vanishing vapor phase. The dotted lines represent the solutions without phase transition. The dashed lines represent the initial conditions: $p_l = 2 \times 10^5$ Pa, $p_r = 10^5$ Pa, $u_l = u_r = 0$ m s$^{-1}$, $Y_l^{liq} = Y_r^{liq} = 0.999$. Final time: $t \approx 8$ ms. Mesh: 100 cells. For the sake of visualization, only 50 symbols out of 100 are plotted for the nonlinear Newton’s method.

given at time of $t \approx 0.3$ ms in Figure 12. The vapor expands and gets colder but remains vapor, as expected.

6.6. Subcooled liquid

Let us now mirror the previous test and consider a pure liquid initially at 350 K corresponding to a subcooled liquid. The pressure is still initially set at $10^5$ Pa, and the initial velocities are unchanged (double expansion). The results are given at two different times, $t \approx 0.165$ ms in Figure 13 and $t \approx 0.65$ ms in Figure 14, exhibiting an interesting behavior.

In Figure 13, fast expansion waves are emitted and propagate into the pure liquid at the sound speed of this phase. They induce pressure drop from 0.1 to 0.04 MPa, corresponding to the saturation pressure at the current temperature. This temperature is shown in the same figure and is nearly constant in the domain, except at the center. This unphysical (but small) temperature increase is well known with all conservative methods and is related to the kinetic energy computation, that is, badly computed from the momentum equation (for example, Cocchi et al. (1998) [25]). Because of this slight temperature increase and pressure drop, a slight variation of the liquid mass fraction appears.

From the expanded liquid at saturation pressure (0.04 MPa), expansion continues as the right and left states still have expansion motion. Two new expansion waves now appear and are clearly visible in Figure 14. They induce a pressure jump from 0.04 to 0.02 MPa. These waves are evaporation fronts (Saurel et al. (2008) [6]) that propagate at $u \pm c_{eq}$, $c_{eq}$ being the mixture sound speed at
Figure 12. Comparison of the present relaxation algorithm (thick lines) versus the nonlinear Newton’s method (symbols). Double expansion test with superheated vapor. The dotted lines represent the solutions without phase transition. The dashed lines represent the initial conditions: $p_l = 10^5$ Pa, $p_r = 10^5$ Pa, $u_l = -200$ m s$^{-1}$, $u_r = 200$ m s$^{-1}$, $Y_l^{liq} = Y_r^{liq} = 0$. Final time: $t \approx 0.3$ ms. Mesh: 100 cells. For the sake of visualization, only 50 symbols out of 100 are plotted for the nonlinear Newton’s method. The vapor expands but remains vapor.

equilibrium, lower than the sound speed given by Eq. (12). Now, evaporation is significant, as about 4% of the liquid mass evaporates.

From these initial conditions of subcooled liquid in expansion, five different waves are emitted as shown in Figure 15.

Four of these waves are clearly visible in Figures 13 and 14 as they correspond to the expansion waves already discussed. The contact wave propagating at the velocity $u = 0$ m s$^{-1}$ is not visible for symmetry reasons. These results confirm the observations of Saurel et al. (2008) [6] regarding evaporation front dynamics in flashing flows. They also explain the kinetic relation of Chaves (1984) [26], which considers evaporation fronts as CJ deflagration waves. They just correspond to expansion waves of the equilibrium model.

7. COMPUTATIONAL TIME, EFFICIENCY, AND SIMPLICITY

In most computational examples considered previously, the computational times of the relaxation solver and the nonlinear Newton’s method are comparable. When dealing with test problems where a pure (or nearly pure) liquid is present, the relaxation algorithm is much faster. For example, Table II displays the different runtime gains for the tests shown in Figures 8 and 13.
Figure 13. Comparison of the present relaxation algorithm (thick lines) versus the nonlinear Newton’s method (symbols). Double expansion test with subcooled liquid. Results are shown at early times ($t \approx 0.165 \text{ ms}$) showing two fast expansion waves emitted into the pure liquid and propagating at the liquid sound speed ($u \pm c_{l,\text{liq}}$). The next figure shows the evolution at longer times, where two additional waves (evaporation fronts) are emitted. The dotted lines represent the solutions without phase transition. The dashed lines represent the initial conditions: $p_l = 10^5 \text{ Pa}$, $p_r = 10^5 \text{ Pa}$, $u_l = -200 \text{ m s}^{-1}$, $u_r = 200 \text{ m s}^{-1}$, $Y_l^{\text{liq}} = Y_r^{\text{liq}} \approx 1$. Mesh: 100 cells. For the sake of visualization, only 50 symbols out of 100 are plotted for the nonlinear Newton’s method.

The main argument with the present method is consequently more related to its simplicity: The relaxation solver presents a straightforward implementation, whereas an iterative method requires solving a nonlinear-algebraic model that may cause difficulties as a result of nonlinearities and single-phase bounds of pure liquid and pure vapor, where it becomes ill posed. Besides, the root-finding Newton’s method requires the calculation of the saturation pressure via Eq. (5) at each iterative step, which itself requires an iterative method.

The reliability of the present method is quite independent of the mesh resolution or the order of accuracy of the numerical scheme. For instance, Figure 16 repeats the test case already presented in Figure 9, with a 10-cell mesh and first-order Godunov numerical scheme. These results are very reasonable, proving robustness of the method. Results for the same test case are also given in Figure 16 for a higher-order solver (MUSCL-Hancock scheme with the Superbee limiter), on a 1000-cell mesh. Again, perfect agreement is found between the present relaxation solver and the nonlinear Newton’s method.

Moreover, the present method easily extends to more than one gas constituent, for example when a non-condensable gas is present in addition to the vapor. Such extension is possible with the iterative Newton’s method, but difficult.
Figure 14. Evolution of the test case of Figure 13 at longer times (here, $t \approx 0.65\, \text{ms}$). In addition to the first two expansion waves propagating with the liquid sound speed ($u \pm c_{\text{liq}}$), two evaporation fronts appear, propagating with the liquid/gas mixture sound speed at equilibrium ($u \pm c_{\text{eq}}$).

Figure 15. Schematic representation in a $(x,t)$ diagram of the five waves present in Figures 13 and 14.

8. MULTI-DIMENSIONAL ILLUSTRATIONS

In this section, the capabilities of the flow model are illustrated on an evaporating liquid jet configuration, reminiscent of coaxial liquid jets, in conditions typical of cryotechnic rocket engines at ignition. The flow contains liquid oxygen and vapor oxygen at cryogenic temperatures. The stiffened gas parameters for oxygen are given in Table III.

The 2D computations are carried out on a simplified geometry given in Figure 17.
A SIMPLE PHASE TRANSITION RELAXATION SOLVER FOR LIQUID–VAPOR FLOWS

Table II. Comparison of the average computational time and gain for the tests shown in Figures 8 and 13.

<table>
<thead>
<tr>
<th>Test</th>
<th>Relaxation solver runtime</th>
<th>Newton’s method runtime</th>
<th>Runtime gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 8</td>
<td>0.95</td>
<td>5%</td>
<td></td>
</tr>
<tr>
<td>Figure 13</td>
<td>0.50</td>
<td>50%</td>
<td></td>
</tr>
</tbody>
</table>

Meshes of 500, 1000, 1500, and 2000 cells were used and yielded the same conclusions.

Figure 16. Comparison of the present relaxation algorithm (thick lines) versus the nonlinear Newton’s method (symbols). Shock tube test case with a two-phase mixture involving a vanishing liquid phase, computed on coarse (left figure) and fine (right figure) meshes for assessing method robustness. The dashed lines represent the initial conditions: $p_l = 2 \times 10^5$ Pa, $p_r = 10^5$ Pa, $u_l = u_r = 0$ m s$^{-1}$, $Y_{liq}^l = Y_{liq}^r = 0.01$. Final time: $t \approx 0.5$ ms. The figure on the left represents the solutions with a 10-cell mesh and first-order Godunov numerical scheme. The figure on the right represents the solutions with a 1000-cell mesh and MUSCL-Hancock scheme with Superbee limiter. Only 50 symbols out of 1000 are plotted for the nonlinear Newton’s method in the figure on the right. The dotted lines represent the solution given in Figure 9 with a 100-cell mesh and first-order Godunov numerical scheme. The relaxation method and iterative one tend to the same solution even with the coarse mesh.

Table III. Stiffened gas coefficients for oxygen.

<table>
<thead>
<tr>
<th>Coefficients</th>
<th>Liquid phase</th>
<th>Vapor phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_p$ (J/kg/K)</td>
<td>1702</td>
<td>780</td>
</tr>
<tr>
<td>$C_v$ (J/kg/K)</td>
<td>695</td>
<td>531</td>
</tr>
<tr>
<td>$y$</td>
<td>2.45</td>
<td>1.47</td>
</tr>
<tr>
<td>$P_{\infty}$ (bar)</td>
<td>1062</td>
<td>0</td>
</tr>
<tr>
<td>$q$ (J/kg)</td>
<td>$-258000$</td>
<td>6900</td>
</tr>
<tr>
<td>$q'$ (J/kg/K)</td>
<td>0</td>
<td>$-9280$</td>
</tr>
</tbody>
</table>

The inlet boundary conditions correspond to two subsonic inflows:

- central flow made of pure liquid oxygen, at 100 K and 30 m s$^{-1}$, pressure 1 MPa along segment AB of Figure 17.
- peripheral flow made of pure gaseous oxygen, at 150 K and 200 m s$^{-1}$, with the pressure of 1 MPa along segment DE of Figure 17.

At the outlet, along segment IH, a non-reflecting subsonic boundary condition is imposed at 1 MPa. The remaining walls are treated as symmetric boundary conditions. The mesh consists of 360,000 triangles, with a spatial discretization varying from 0.1 mm in the jet to 0.6 mm in the far field.

Copyright © 2016 John Wiley & Sons, Ltd.  
DOI: 10.1002/fld
The small scale destabilization of the liquid jet requires to extend the numerical solver presented in Saurel et al. (2016) [13] to second order. This was achieved following the MUSCL-Hancock scheme with the Superbee limiter, as detailed for example in Toro (1997) [24].

Two computations have been carried out on the same mesh, with the aim of studying the effect of the thermochemical relaxation on the jet destabilization. The contours of mixture density as obtained with and without evaporation effects are compared in Figure 18.

The filaments separating the main liquid core and the gas gradually vanish as a consequence of evaporation. Figure 19 compares the corresponding temperature contours. The thermochemical relaxation solver has the expected behavior: Pure liquid oxygen and pure gas oxygen are free to
evolve out of the saturation conditions (resp. at 100 and 150 K), whereas $T = T_{\text{sat}}(p)$ where the
flow consists of a two-phase mixture.

For the configuration tested here, the CPU saving of the present algorithm over the iterative
method is of the order of 18\%, with no noticeable change in the results.

In future works, the thermochemical relaxation solver will be extended to take into account a mul-
ticomponent gas mixture ($\text{H}_2$, He, $\text{H}_2\text{O}$, etc.) which is present in many applications. Also, capillary
effects are not included in these simulations, as the intense velocity gradients make them negligi-
ble in the present jet destabilization conditions. However, capillary effects may be of importance
when the filaments start separating from the jet, and our future work will include them, following
the strategy presented in Le Martelot et al. (2014) [14].

9. CONCLUSIONS

A simple relaxation solver has been built to compute phase equilibrium in multiphase compressible
flows. Computational examples have shown that the method converges to the same solution as meth-
ods based on iterative (and exact) equilibrium solvers. The new method uses two main ingredients
to estimate the liquid (or vapor) equilibrium mass fraction:

- Minmod-type treatment of the source term, and
- an accurate estimate of the equilibrium pressure based on some linearization of the equilibrium
  mass fraction sources as functions of the pressure.

The method also considers single-phase limits made of pure liquid and pure vapor. The next step
will be to address gas mixtures in conjunction with a liquid phase, in the direction of two-phase
combustion.

Copyright © 2016 John Wiley & Sons, Ltd.

DOI: 10.1002/fld
APPENDIX A.

A1. Analytical expressions for the iterative algorithm

Based on the expressions of $Y_m^l(p^*)$ and $Y_e^l(p^*)$ from Eqs. (21) and (22), let us introduce the following function:

$$f(p) = \left[ e_l(p) - e_g(p) \right] \left[ v_l(p) - v_g(p) \right] - \left[ e_l(p) - e_g(p) \right] \left[ v_l(p) - v_g(p) \right],$$  \hspace{1cm} (A.1)

whose solution is the exact equilibrium pressure obtained with Newton’s iteration,

$$p^{n+1} = p^n - \frac{f(p)}{f'(p)},$$  \hspace{1cm} (A.2)

The derivative of $f$ with respect to the pressure is given by

$$f'(p) = \frac{df}{dp} = \left( \frac{\partial f}{\partial p} \right)_T + \left( \frac{\partial f}{\partial T} \right)_p \frac{dT}{dp},$$  \hspace{1cm} (A.3)

$\left( \frac{\partial f}{\partial p} \right)_T$ and $\left( \frac{\partial f}{\partial T} \right)_p$ are straightforward to derive analytically. In the case $p \ll p_{\infty,l}$, they reduce to,

$$\left( \frac{\partial f}{\partial p} \right)_T = \left( \gamma_g - 1 \right) C_{v,g} T \left( \gamma_l C_{v,l} T - e + q_l \right) \frac{p^2}{p^2},$$  \hspace{1cm} (A.4)

$$\left( \frac{\partial f}{\partial T} \right)_p = C_{v,g} \left( -2 \gamma_l \left( \gamma_g - 1 \right) C_{v,l} T + \left( \gamma_g - 1 \right) (e - q_l) \right) + \left( \gamma_l C_{v,l} - C_{v,g} \right) p v.$$  \hspace{1cm} (A.5)

To compute $\frac{dT}{dp}$, let us recall the saturation temperature,

$$T = T_{\text{sat}}(p) = \frac{B}{C W \left( -\frac{B e^{A/C} p^{-1/C} (p + p_{\infty,l})^{D/C}}{C} \right)},$$  \hspace{1cm} (A.6)

where $W$ is the Lambert W function. Differentiating the previous function and making use of the Lambert W function lead to,

$$\frac{dT_{\text{sat}}}{dp} = \frac{B ((D - 1) p - p_{\infty,l})}{C^2 p (p + p_{\infty,l}) W \left( -\frac{B e^{A/C} p^{-1/C} (p + p_{\infty,l})^{D/C}}{C} \right) W \left( -\frac{B e^{A/C} p^{-1/C} (p + p_{\infty,l})^{D/C}}{C} \right) + 1)},$$  \hspace{1cm} (A.7)

which can be conveniently expressed as

$$\frac{dT_{\text{sat}}}{dp} = \frac{T_{\text{sat}}^2 ((D - 1) p - p_{\infty,l})}{p (p + p_{\infty,l}) (B - CT_{\text{sat}})}.$$  \hspace{1cm} (A.8)

ACKNOWLEDGEMENTS

Part of this work has been carried out in the framework of the Labex MEC (ANR-10-LABX-0092) and of the A*MIDEX project (ANR-11-IDEX-0001-02), funded by the Investissements d’Avenir French Government program managed by the French National Research Agency (ANR). We also acknowledge funding from ANR through project ANR-14-CE22-0014.
REFERENCES

Research Article

A simple phase transition relaxation solver for liquid-vapor flows

Alexandre Chiapolino, Pierre Boivin, and Richard Saurel

This work presents a new method to compute thermochemical equilibrium in liquid–vapor flows. The proposed method is both accurate and fast, in addition of being much easier to code than the usual iterative methods. Through a series of test cases from simple 1D flow configurations to a complex 2D evaporating liquid jet, the solver is proved to successfully cope with cavitation, evaporation, condensation, and boiling.

Wiley Online Library Graphical TOC
Dear Author,

During the copyediting of your paper, the following queries arose. Please respond to these by annotating your proof with the necessary changes/additions.

- If you intend to annotate your proof electronically, please refer to the E-annotation guidelines.
- If you intend to annotate your proof by means of hard-copy mark-up, please use the standard proofreading marks in annotating corrections. If manually writing corrections on your proof and returning it by fax, do not write too close to the edge of the paper. Please remember that illegible mark-ups may delay publication.

Whether you opt for hard-copy or electronic annotation of your proof, we recommend that you provide additional clarification of answers to queries by entering your answers on the query sheet, in addition to the text mark-up.

<table>
<thead>
<tr>
<th>Query No.</th>
<th>Query</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>AUTHOR: Please check that the authors and their affiliations are correct.</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>AUTHOR: Please provide affiliation address (city and country) for University Institute of France.</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>AUTHOR: Please define NASG if it is an acronym.</td>
<td></td>
</tr>
<tr>
<td>Q4</td>
<td>AUTHOR: Please define HLLC if it is an acronym.</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>AUTHOR: Please define CJ if it is an acronym.</td>
<td></td>
</tr>
<tr>
<td>Q6</td>
<td>AUTHOR: Please define MUSCL if it is an acronym.</td>
<td></td>
</tr>
<tr>
<td>Q7</td>
<td>AUTHOR: Please define DE if it is an acronym.</td>
<td></td>
</tr>
<tr>
<td>Q8</td>
<td>AUTHOR: Please define IH if it is an acronym.</td>
<td></td>
</tr>
<tr>
<td>Q9</td>
<td>AUTHOR: Please provide city and country location of the publisher for Reference 2, 15, and 24.</td>
<td></td>
</tr>
<tr>
<td>Q10</td>
<td>AUTHOR: If this is not a one-page article please supply the first and last pages for this article for Reference 22.</td>
<td></td>
</tr>
</tbody>
</table>
1. Corrections should be marked with the Adobe Annotation & Comment Tools below:

2. To save your proof corrections, click the ‘Publish Comments’ button. Publishing your comments saves the marked up version of your proof to a centralized location in Wiley’s Online Proofing System. Corrections don’t have to be marked in one sitting – you can publish corrections and log back in at a later time to add more.

3. When your proof review is complete we recommend you download a copy of your annotated proof for reference in any future correspondence concerning the article before publication. You can do this by clicking on the icon to the right of the ‘Publish Comments’ button and selecting ‘Save as Archive Copy’.

4. When your proof review is complete and you are ready to send corrections to the publisher click the ‘Complete Proof Review’ button that appears above the proof in your web browser window. Do not click the ‘Complete Proof Review’ button without replying to any author queries found on the last page of your proof. Incomplete proof reviews will cause a delay in publication. Note: Once you click ‘Complete Proof Review’ you will not be able to mark any further comments or corrections.

Firefox, Chrome, Safari Users

If your PDF article proof opens in any PDF viewer other than Adobe Reader or Adobe Acrobat, you will not be able to mark corrections and query responses, nor save them. To mark and save corrections, please follow these instructions to disable the built-in browser PDF viewers in Firefox, Chrome, and Safari so the PDF article proof opens in Adobe within a Firefox, Chrome, or Safari browser window.